

데이터베이스

관계형 데이터베이스

05. 그룹 함수

first
coding

01 그룹 함수



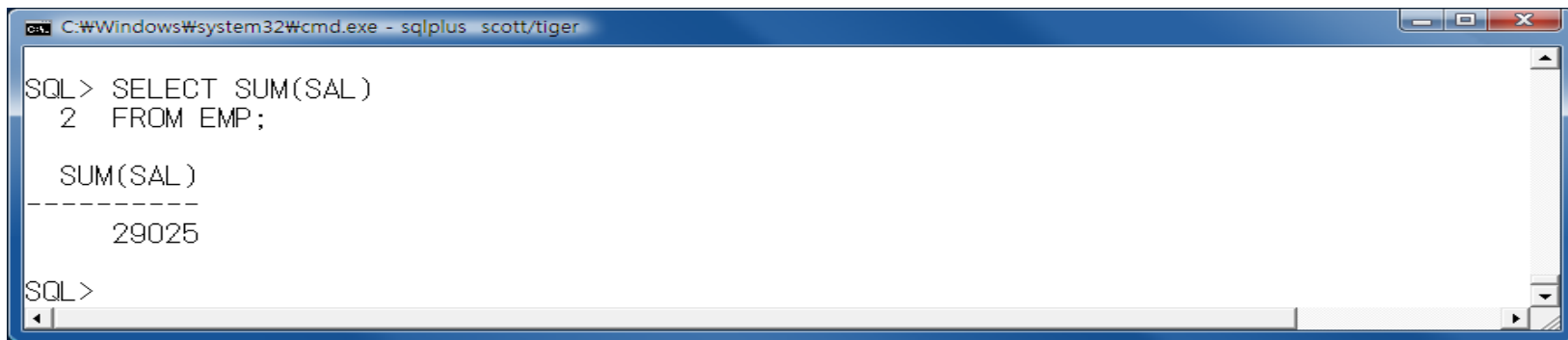
- 그룹 함수는 **하나 이상의 행을** 그룹으로 묶어 연산하여 총합, 평균 등 하나의 결과로 나타냅니다.
- 다음은 그룹 함수의 종류를 정리한 표입니다.

구 분	설 명
<u>SUM</u>	그룹의 누적 합계를 반환합니다.
<u>AVG</u>	그룹의 평균을 반환합니다.
<u>COUNT</u>	그룹의 총 개수를 반환합니다.
<u>MAX</u>	그룹의 최대값.을 반환합니다.
<u>MIN</u>	그룹의 최소값.을 반환합니다.
STDDEV	그룹의 표준편차를 반환합니다.
VARIANCE	그룹의 분산을 반환합니다.

- SUM 함수는 해당 칼럼 값들에 대한 총합을 구하는 함수입니다.
- 다음은 급여 총액 구하는 예제입니다.

예

```
SELECT SUM(SAL)
FROM EMP;
```



The screenshot shows a Windows command prompt window titled "C:\Windows\system32\cmd.exe - sqlplus scott/tiger". Inside the window, the following SQL query is entered and executed:

```
SQL> SELECT SUM(SAL)
2  FROM EMP;
```

The output of the query is displayed as follows:

```
SUM(SAL)
-----
29025
```

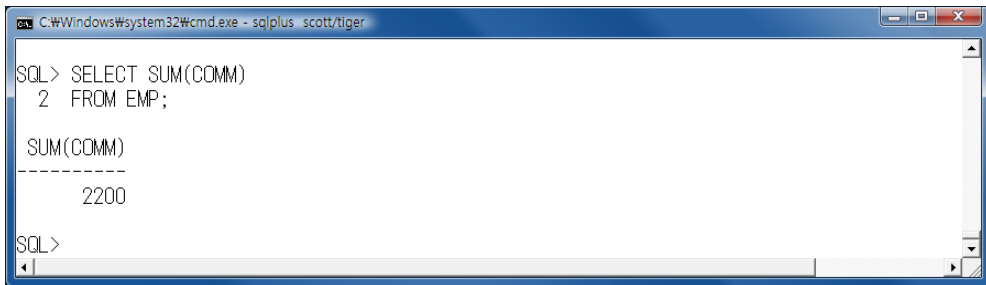
The prompt "SQL>" is visible at the bottom of the window.

- 다음은 커미션 총액 구하는 예제입니다.

NULL

예

```
SELECT SUM(COMM)  
FROM EMP;
```



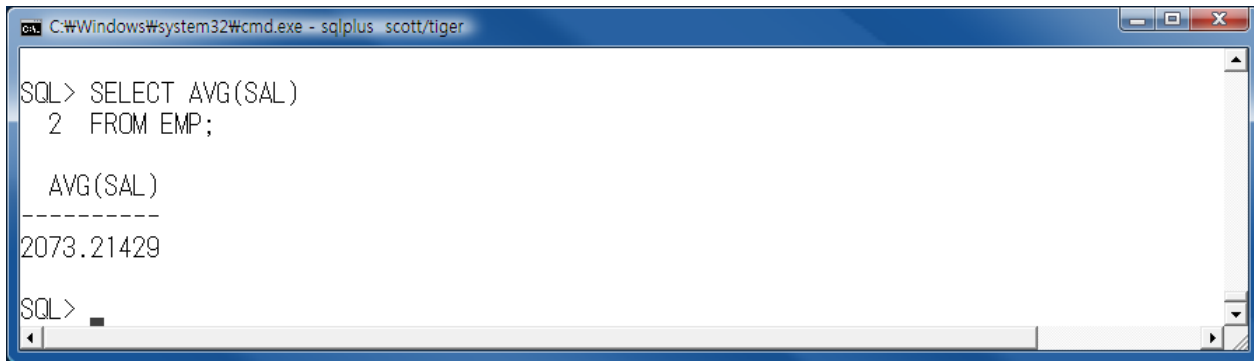
```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger  
  
SQL> SELECT SUM(COMM)  
2  FROM EMP;  
  
SUM(COMM)  
-----  
2200  
  
SQL>
```

- 사원 테이블을 살펴보면 커미션 컬럼에 NULL 값이 저장된 사원이 존재합니다.
- NULL 은 블랙 홀이므로 NULL을 저장한 컬럼과 연산한 결과도 NULL 이라고 하였습니다.
- 커미션의 총합을 구해도 NULL 값으로 출력되지 않습니다.
- 왜냐하면 그룹 함수는 다른 연산자와는 달리 해당 칼럼 값이 NULL 인 것을 제외하고 계산하기 때문입니다.

- AVG 함수는 해당 칼럼 값들에 대해 평균을 구하는 함수입니다. 이때 해당 칼럼 값이 NULL 인 것에 대해서는 제외하고 계산합니다.
- 다음은 급여 평균 구하는 예제입니다.

예

```
SELECT AVG(SAL)
FROM EMP;
```



```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT AVG(SAL)
2  FROM EMP;

AVG(SAL)
-----
2073.21429

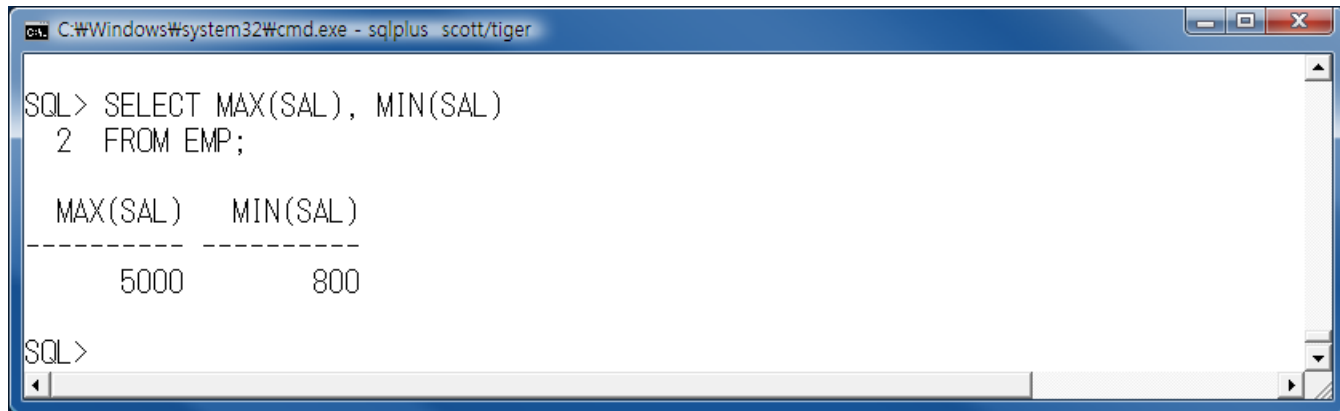
SQL>
```

01 3. 최대값 구하는 MAX, 최소값 구하는 MIN 함수

- 지정한 칼럼 값들 중에서 최대값을 구하는 함수가 MAX이고, 최소값을 구하는 함수가 MIN입니다.
- 다음은 가장 높은 급여와 가장 낮은 급여를 구하는 예제입니다.

예

```
SELECT MAX(SAL), MIN(SAL)
FROM EMP;
```



```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT MAX(SAL), MIN(SAL)
2  FROM EMP;

  MAX(SAL)    MIN(SAL)
-----
      5000         800

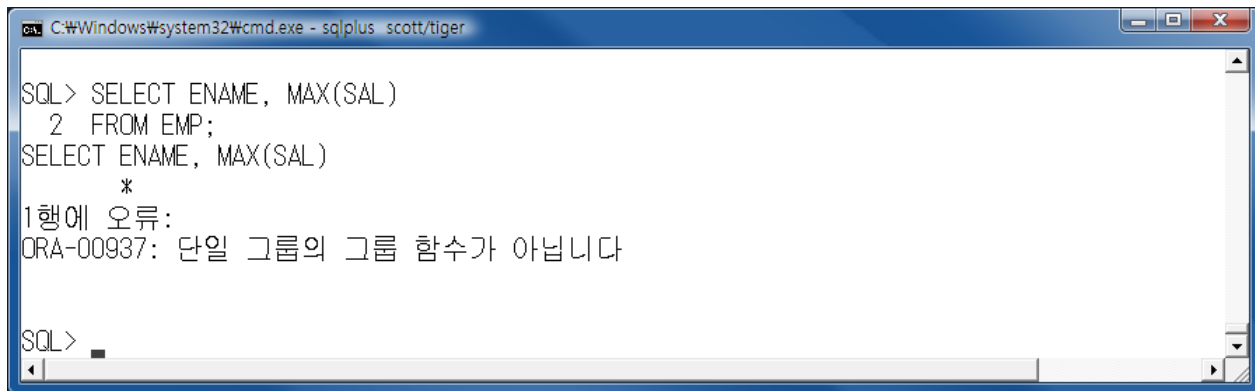
SQL>
```

- 직원들의 최대 급여는 다음과 같이 구해집니다.

예

```
SELECT MAX(SAL)
FROM EMP;
```

- 위 쿼리문의 SELECT 절에 ENAME 컬럼만 추가한다고 될까요? 아래와 같이 사원의 이름도 함께 출력해 봅니다.



```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

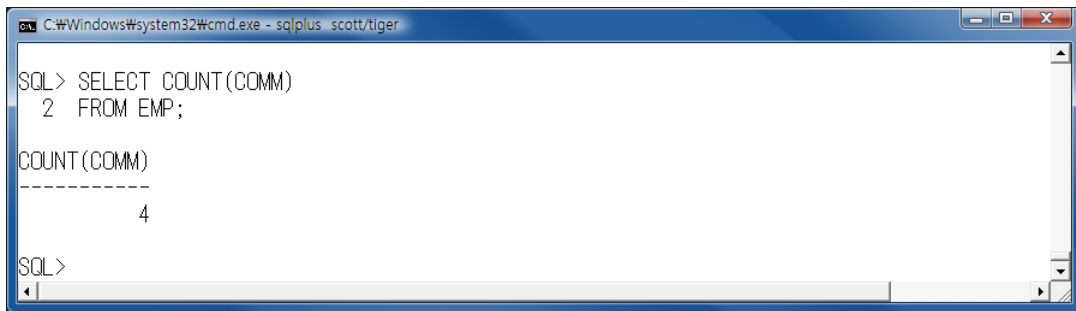
SQL> SELECT ENAME, MAX(SAL)
      2 FROM EMP;
SELECT ENAME, MAX(SAL)
      *
1행에 오류:
ORA-00937: 단일 그룹의 그룹 함수가 아닙니다

SQL>
```

- COUNT 함수는 테이블에서 조건을 만족하는 행의 개수를 반환하는 함수입니다. COUNT 함수에 특정 칼럼을 기술하면 해당 칼럼 값을 갖고 있는 로우의 개수를 계산하여 되돌립니다.
- 다음은 사원 테이블의 사원들 중에서 커미션을 받은 사원의 수를 구하는 예제입니다.

예

```
SELECT COUNT(COMM)  
FROM EMP;
```

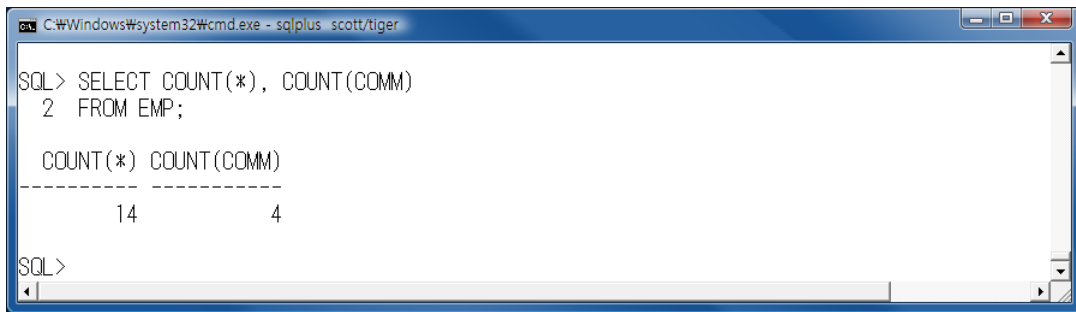


```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger  
  
SQL> SELECT COUNT(COMM)  
2 FROM EMP;  
  
COUNT(COMM)  
-----  
4  
  
SQL>
```


- COUNT 함수는 NULL 값에 대해서는 개수를 세지 않습니다.
- 만일 COUNT 함수에 COUNT(*)처럼 *를 적용하면 테이블의 전체 로우 수를 구하게 됩니다.
- 전체 사원의 수와 커미션을 받는 사원의 수를 구하는 예제입니다.

예

```
SELECT COUNT(*), COUNT(COMM)
FROM EMP;
```



The screenshot shows a Windows command prompt window titled "C:\Windows\system32\cmd.exe - sqlplus scott/tiger". Inside the window, the following SQL query is entered and executed:

```
SQL> SELECT COUNT(*), COUNT(COMM)
2 FROM EMP;
```

The output of the query is displayed as follows:

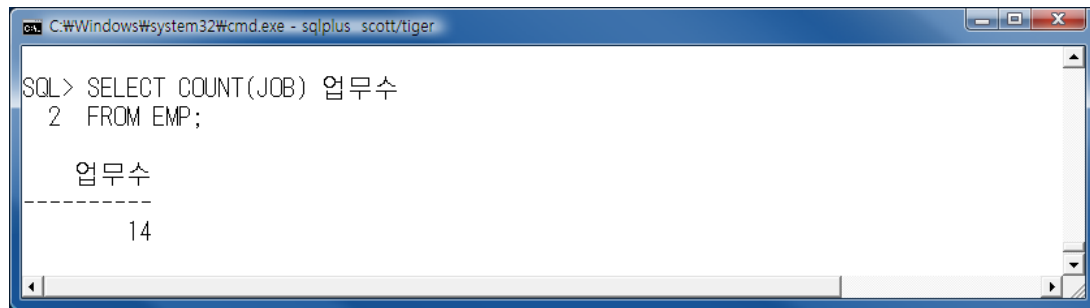
COUNT(*)	COUNT(COMM)
14	4

The window also shows the prompt "SQL>" at the bottom, indicating the command is ready for the next input.

- 이번에는 사원 테이블에서 사원들의 직업의 개수를 카운트 해봅시다.

예

```
SELECT COUNT(JOB) 업무수  
FROM EMP;
```



The screenshot shows a Windows command prompt window titled "C:\Windows\system32\cmd.exe - sqlplus scott/tiger". Inside the window, the following SQL query is entered and executed:

```
SQL> SELECT COUNT(JOB) 업무수  
2 FROM EMP;
```

The output of the query is displayed as follows:

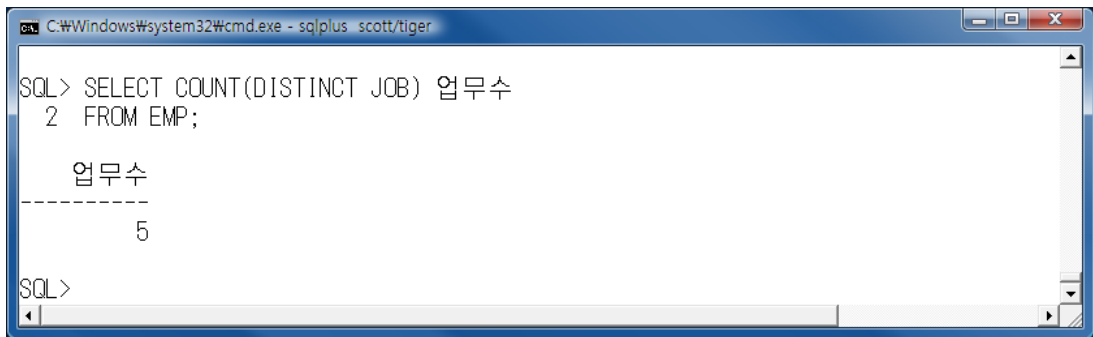
```
      업무수  
-----  
          14
```

01 5. 로우 개수 구하는 COUNT 함수

- 직업의 종류가 몇 개인지 즉, 중복되지 않은 직업의 개수를 카운트 해봅시다.
- 이 때 중복 행 제거 키워드 DISTINCT를 써서 다음과 같이 질의합니다.

예

```
SELECT COUNT(DISTINCT JOB) 업무수  
FROM EMP;
```



The screenshot shows a Windows command prompt window titled "C:\Windows\system32\cmd.exe - sqlplus scott/tiger". Inside the window, the following SQL query is entered and executed:

```
SQL> SELECT COUNT(DISTINCT JOB) 업무수  
2 FROM EMP;
```

The output of the query is displayed as follows:

```
      업무수  
-----  
          5
```

The prompt "SQL>" is visible at the bottom of the window.

- 그룹함수를 쓰되 어떤 컬럼 값을 기준으로 그룹함수를 적용할 경우 GROUP BY 절 뒤에 해당 컬럼을 기술하면 됩니다.

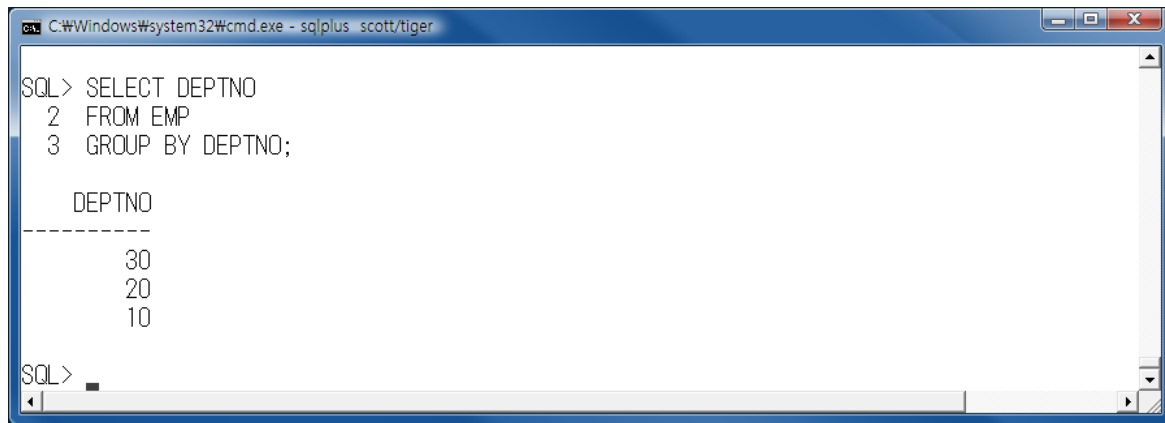
형식	<pre>SELECT 컬럼명, 그룹함수 FROM 테이블명 WHERE 조건 (연산자) GROUP BY 컬럼명;</pre>
----	--

- 합계, 평균, 최대값.이나, 최소값. 등을 어떤 칼럼을 기준으로 그 칼럼의 값 별로 보고자 할 때 GROUP BY 절 뒤에 해당 칼럼을 기술하면 됩니다.
- GROUP BY 절을 사용할 때 주의할 점은 GROUP BY 절 다음에는 칼럼의 별칭을 사용할 수 없고, 반드시 칼럼명을 기술해야 한다는 점입니다.

- 사원 테이블을 부서 번호로 그룹지어 봅시다.

예

```
SELECT DEPTNO  
FROM EMP  
GROUP BY DEPTNO;
```



The screenshot shows a Windows command prompt window titled "C:\Windows\system32\cmd.exe - sqlplus scott/tiger". The user has entered the following SQL command:

```
SQL> SELECT DEPTNO  
2 FROM EMP  
3 GROUP BY DEPTNO;
```

The output of the query is displayed as follows:

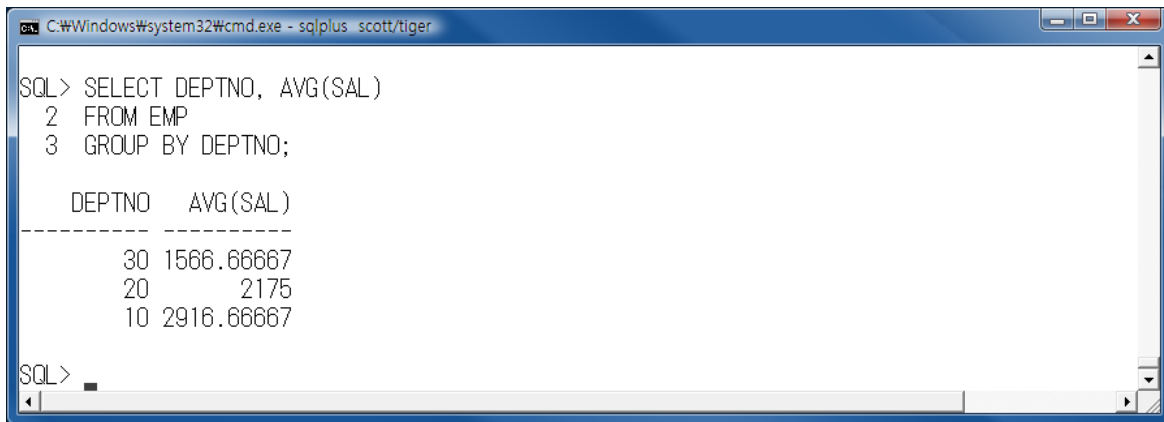
```
DEPTNO  
-----  
30  
20  
10
```

The window shows the command prompt history and the output of the query. The cursor is at the bottom of the window, ready for the next command.

- 다음은 소속 부서별 평균 급여 구하는 예제입니다.

예

```
SELECT DEPTNO, AVG(SAL)
FROM EMP
GROUP BY DEPTNO;
```



The screenshot shows a Windows command prompt window titled "C:\Windows\system32\cmd.exe - sqlplus scott/tiger". Inside the window, the following SQL query is entered and executed:

```
SQL> SELECT DEPTNO, AVG(SAL)
2  FROM EMP
3  GROUP BY DEPTNO;
```

The output of the query is displayed as a table with two columns: DEPTNO and AVG(SAL). The data is as follows:

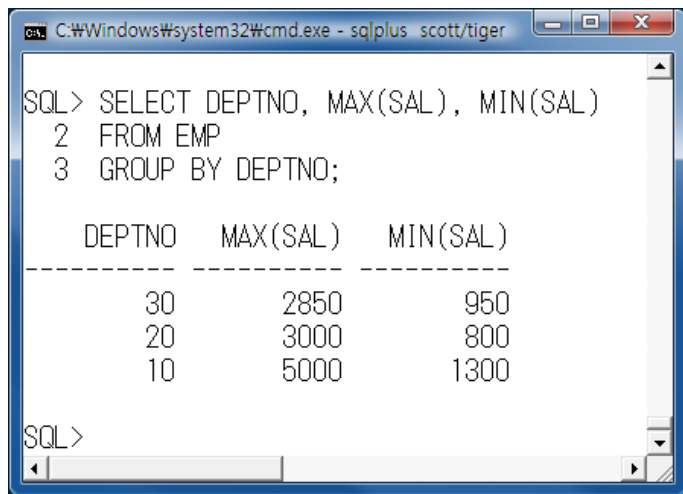
DEPTNO	AVG(SAL)
30	1566.66667
20	2175
10	2916.66667

The command prompt shows the prompt "SQL>" at the bottom, indicating the query has been executed.

- 다음은 소속 부서별 최대 급여와 최소 급여를 구하는 예제입니다.

예

```
SELECT DEPTNO, MAX(SAL), MIN(SAL)
FROM EMP
GROUP BY DEPTNO;
```



The screenshot shows a Windows command prompt window titled "C:\Windows\system32\cmd.exe - sqlplus scott/tiger". The user has entered the following SQL query:

```
SQL> SELECT DEPTNO, MAX(SAL), MIN(SAL)
2 FROM EMP
3 GROUP BY DEPTNO;
```

The output of the query is displayed in a table format:

DEPTNO	MAX(SAL)	MIN(SAL)
30	2850	950
20	3000	800
10	5000	1300

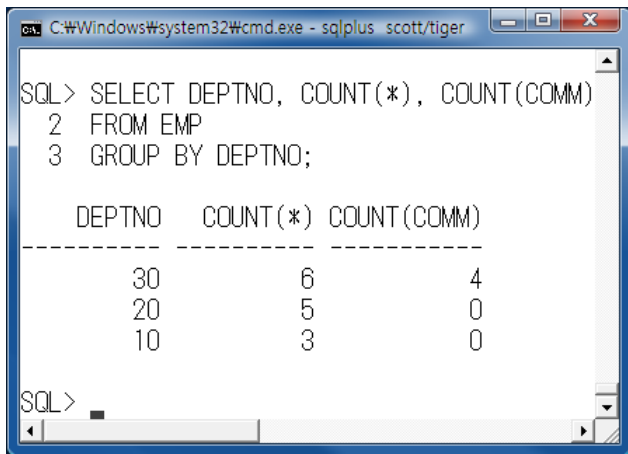
The prompt "SQL>" is visible at the bottom of the window.

02 GROUP BY 절

- 다음은 부서별로 직원 수와 커미션을 받는 직원들의 수를 계산하는 예제입니다.

예

다음은 부서별로 직원 수와 커미션을 받는 직원들의 수를 계산하는 예제입니다.



```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT DEPTNO, COUNT(*), COUNT(COMM)
2  FROM EMP
3  GROUP BY DEPTNO;

   DEPTNO   COUNT(*)  COUNT(COMM)
-----
       30         6          4
       20         5          0
       10         3          0

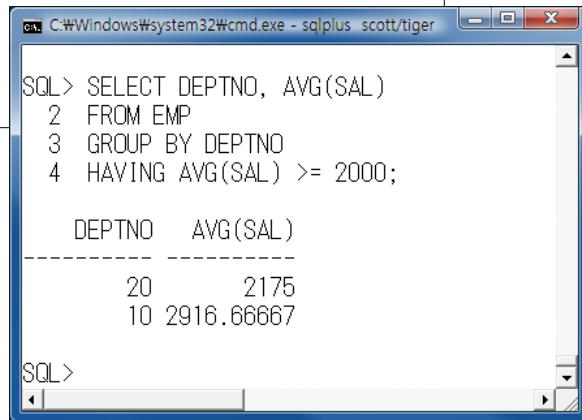
SQL>
```

```
SELECT deptno, count(*), count(comm)
from emp
group by deptno;
```


- SELECT 절에 조건을 사용하여 결과를 제한할 때는 WHERE 절을 사용하지만 그룹의 결과를 제한할 때는 HAVING 절을 사용합니다.
- 예를 들어 설명하자면 부서별로 그룹지은 후(GROUP BY), 그룹 지어진 부서별 평균 급여가 2000 이상인(HAVING) 부서번호와 부서별 평균 급여를 출력하는 경우입니다.

예

```
SELECT DEPTNO, AVG(SAL)
FROM EMP
GROUP BY DEPTNO
HAVING AVG(SAL) >= 2000;
```



The screenshot shows a Windows command prompt window titled "C:\Windows\system32\cmd.exe - sqlplus scott/tiger". The prompt is "SQL>". The user has entered the following SQL query:

```
SQL> SELECT DEPTNO, AVG(SAL)
2 FROM EMP
3 GROUP BY DEPTNO
4 HAVING AVG(SAL) >= 2000;
```

The query results are displayed as follows:

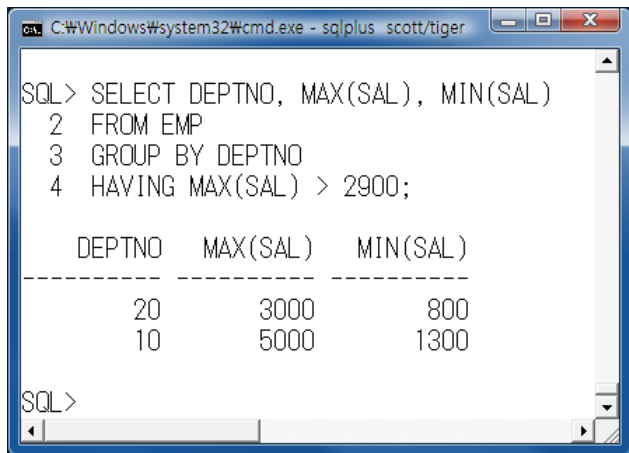
DEPTNO	AVG(SAL)
20	2175
10	2916.66667

The prompt "SQL>" is visible at the bottom of the window.

- 부서의 최대값과 최소값을 구하되 최대 급여가 2900이상인 부서만 출력합니다.

예

```
SELECT DEPTNO, MAX(SAL), MIN(SAL)
FROM EMP
GROUP BY DEPTNO
HAVING MAX(SAL) > 2900;
```



```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT DEPTNO, MAX(SAL), MIN(SAL)
2  FROM EMP
3  GROUP BY DEPTNO
4  HAVING MAX(SAL) > 2900;

   DEPTNO   MAX(SAL)   MIN(SAL)
-----
        20        3000         800
        10        5000        1300

SQL>
```